REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE August, ¢997	3. REPORT TYPE AND DATES COVERED Final

4. TITLE AND SUBTITLE

The Voive-Activated Multilingual Interview System

5. FUNDING NUMBERS

MDA972-96-C-0007

C

6. AUTHOR(S)

Paul G. Bamberg
Carol Kunz

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Dragon Systems, Inc.
320 Nevada Street
Newton, MA 02160

8. PERFORMING ORGANIZATION REPORT NUMBER

Final Technical Report

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Defense Advanced Research Projects Agency
ATTN: Dr. J. Allen Sears
3701 N. Fairfax Drive
Arlington, VA 22203

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

CLIN 0007

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT

Distribution Statement A - Distribution is unlimited

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

The Multilingual Interview System is a Windows-based application program designed to let users conduct simple interviews by voice in languages they do not speak. Any statement or question that is within the vocabulary, when spoken into a microphone attached to the computer, is recognized by a large-vocabulary speech recognition system and converted into a sequence of pre-recorded wave files which are then played back through a loudspeaker attached to the computer. The development of the operational applications for this system was done by NOMI in Pensacola, Florida. Dragon Systems, Inc. developed and customized the underlying speech recognition technology and produced the application softwarethat connects the spoken input to the spoken output as well as the software to record foreign-language wave files for playback. The system was tested at Ft. Bragg and deloyed in Bosnia on laptop, handheld and wearable computers.

14. SUBJECT TERMS

MIS
Speech Recognition

15. NUMBER OF PAGES

29

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UL

Final Technical Report

The Voice-Activated Multilingual Interview System

Dr. Paul G. Bamberg, Principal Investigator and Author
Ms. Carol Kunz, Co-Author
Dr. Patri J. Pugliese, Editor

Dragon Systems, Inc.
320 Nevada Street
Newton, Massachusetts 02160

The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Defense Advanced Research Projects Agency or the U.S. Government.

Distribution Statement A - Distribution is unlimited

Sponsored by:
Defense Advanced Research Projects Agency
Information Technology Office
DARPA Order No. D773
Program Code No.: 5E20
Issued by DARPA/CMO under Contract #MDA972-96-C-0007

Dragon Systems, Inc.
Newton, Massachusetts
August, 1997

ABSTRACT

The Multilingual Interview System is a Windows-based application program for a laptop, handheld, or wearable personal computer whose intent is to let users conduct simple interviews by voice in languages that they do not speak. To configure the application, the user selects a domain, such as "force protection" or "mine awareness and training" and an output language, such as Croatian or Haitian Creole. Any statement or question that is within the vocabulary for the selected domain, when spoken into a microphone attached to the computer, is recognized by a large-vocabulary speech recognition system and converted into a sequence of pre-recorded wave files which are then played back through a loudspeaker attached to the computer. In the case where a question was spoken, the application may be configured to bring up a dialog to aid in eliciting a simple response such as "yes-no" or a number from the subject of the interview. These responses, along with what was spoken by the interviewer, can be saved in a text file which provides a complete record of the interview.

The strength of this system lies in its simplicity. Since it relies only on isolated-word technology (a highly customized version of the DragonDictate product), it can perform with high accuracy even on an 80486 processor. Since it uses files recorded by native speakers rather than synthesized speech, the voice output sounds natural to the person being interviewed. Finally, since the conversion from English to the output language is done by table lookup into spoken phrases that were selected by expert translators, the spoken output is guaranteed to be an accurate translation of the recognized input.

The development of the operational applications for this system was done by NOMI (Naval Operational Medical Institute) in Pensacola, FL. Dragon Systems, Inc. developed and customized the underlying speech recognition technology and produced the application software that connects the spoken input to the spoken output as well as the software that is used to record the foreign-language wave files for playback. The complete system was tested at Fort Bragg and deployed for use by U.S forces in Bosnia, initially on laptop computers and later on handheld and wearable computers.

TABLE OF CONTENTS

PREFACE

The Multilingual Interview System had its origins in the Medical Translator developed by U.S. Navy CDR Lee Morin. This system, developed as a response to the difficulties encountered in communicating with wounded Iraqi POWs during the Gulf War, had an inventory of about 2000 medical words and phrases, whose translations in almost 50 different languages could be played aloud by clicking on the phrase or by typing code numbers into a Visual Basic application.

In January of 1996, Dragon Systems demonstrated a rudimentary voice interface for the Medical Translator. On the basis of this demonstration, the project that is described in this report was begun, with the aim of creating a voice-controlled system with a vocabulary appropriate for use in Bosnia. The initial deployment to Bosnia was on laptop computers, with Dragon speech recognition controlling the original Medical Translator application program, into which NOMI had substituted a new vocabulary.

In October of 1996, the original Visual Basic application software was replaced by a Dragon Systems application, written in C++, with a great deal of added functionality. The target platform for deployment was changed from laptop computers to handheld units created by Carnegie-Mellon University and wearable computers manufactured by CDI. These were used for the most recent deployment to Bosnia in May 1997.

This report will focus on the speech-recognition aspects of the Multilingual Interview System. Equally important to the overall success of the system are the data files and voice recordings that define the content of the modules that were deployed to Bosnia. These were all created at NOMI in Pensacola by a team led by CAPT Michael Valdez and LCDR Eric Rasmussen.

SUMMARY

Task Objectives:

The original goal of this project was to develop a speech interface to a version of the Lee Morin Medical Translator that had been specially modified with a vocabulary appropriate for use by U.S forces in Bosnia and to collaborate with NOMI (Naval Operational Medical Institute) in preparing speech-enabled systems on laptop computers for deployment to Bosnia. The scope of the project was later expanded to include development of new C++ application software to replace the Medical Translator and to deliver deployable systems on handheld and wearable computers.

Technical Problems:

Although the speech-recognition task involved (distinguishing among no more than about 10,000 words, phrases, and sentences, all presumed equally likely) is arguably easier than the 60,000-word dictation task for which the DragonDictate product was developed, three major technical problems arose.

The first problem was to automate the task of replacing the entire dictation vocabulary with a task-specific vocabulary of phrases whose effect was to play wave files. This was accomplished by developing a set of tools to add any required new words and their pronunciations to the 200,000-word Dragon pronouncing dictionary and to generate the required DragonDictate data files from files supplied by NOMI that define the modules to be deployed.

The second problem was to provide voice control of all the features of the application program. Since DragonDictate is designed to track the active window in an application, this problem was solved fairly satisfactorily simply by defining an appropriate active vocabulary for each window in the application program, but thorny arose issues in conjunction with turning the microphone off to permit playback of audio files, then back on to receive the next command. Voice control of modes that permit confirmation by the user or that record a response was also a major challenge, as was keeping track of the appropriate set of phrases to recognize when the user changes the active module. These problems were finally solved only by Dragon's release of a new toolkit.

The third problem was to make recognition accuracy as close to 100% as possible without requiring users to invest more than a few minutes in an enrollment session to adapt the initial speaker-independent acoustic models. This problem has proved unexpectedly difficult, principally because the rapid-match models that must be used to reduce the recognition task to a manageable size for an 80486 processor were trained from isolated words and therefore fare poorly with sentences that begin with sequences like "do you know..."

In addition to these three problems, which were Dragon's alone, there were a host of technical problems that had to be solved jointly by Dragon and NOMI to produce a robust operational system. Many of these were simply a consequence of working with handheld computers that were still under development, but there were also fundamental issues in finding a satisfactory microphone with a push-to-talk switch.

General Methodology:

What motivated this entire project was the realization that the DragonDictate product contained features that would permit rapid customization to the task at hand rapidly enough to permit timely deployment to Bosnia. Therefore the general approach was settled from the start: there was no consideration given to using continuous speech recognition, for example, or to making any fundamental changes in speech recognition algorithms.

The basic methodology was to build and test successively closer and closer approximations to the final deployable system. Each refinement led to more appropriate hardware, a better user interface, and more satisfactory values for several adjustable parameters in the recognizer. Because the recognition engine was already a thoroughly tested product rather than a research system, testing of recognition performance on prerecorded scripts was confined to the two areas – rapid match models and performance with very enrollment – in which the task differed substantially from ordinary isolated-word dictation.

Technical Results:

To our surprise, we learned that near-perfect recognition could be achieved, even under laboratory test conditions, only on Pentium-based computers or on 80486-based computers with a vocabulary that was small enough so that the rapid-match step in the recognition could be bypassed. The speaker-independent hidden Markov models, especially after a small amount of adaptation, gave an error rate of less than 2% on our tests, but the rapid-match algorithm, which for isolated-word dictation can reliably place the correct word in the top 1000 choices out of 30,000, led to error rates as high as 10% on some tests with vocabularies of a few thousand words. Use of a push-to-talk switch, which was almost essential for the high-noise environments encountered under operational conditions, also caused recognition errors, sometimes as a result of transients from the switch itself, sometimes because the user began to speak before pressing the switch.

Given the opportunities for user confirmation built into the overall system, the speech recognition performance was adequate under almost all circumstances, but it was impressively good (98% accuracy or better) only on laptop computers with a good audio front end.

Important Findings and Conclusions:

Although the Multilingual Interview System employs neither continuous speech recognition nor general automatic translation, most users do not seem concerned by this modest level of technology. To users, what turns out to matter is that the system always generates output that sounds good and is very accurate. Even the requirement that the precise wording of phrases had to be memorized did not upset the highly-motivated users. Nonetheless, users seemed to master only a fairly small subset of the very large vocabulary. For that reason, the elaborate support for access to topic-specific phrases in the form of customizable dialogs and keyword searches proved to be an important feature of the system.

On the basis of our experience with development and operational testing of this system, we conclude that the following improvements would most enhance its performance and usability:

1) Eliminate the need for the current rapid-match step in the speech recognition. This can be achieved either by using a sufficiently powerful processor (a fast Pentium) a sufficiently small vocabulary (1000 phrases active per module with an 80486) or a rapid-match algorithm that is based on hidden Markov models.
2) Enlarge the vocabulary of the system so that any reasonable wording of a sentence in the vocabulary will lead to the desired output. At a minimum, the user should be able to substitute "some" for "any" or "are you able to" for "can you" and to omit phrases like "of any kind."
3) When designing the application vocabulary, keep in mind the use of speech recognition and the sequential playback of audio wave files. This principle would eliminate features like large numbers of vocabulary items that begin with the same word sequence like "do you know how to."
4) In selecting hardware for deployment, pay careful attention to the quality of the audio input and the microphone.

Significant Developments:

In speech recognition, this project has led to new techniques for rapid customization of vocabulary and new insights into the performance of systems that use isolated-word techniques to distinguish among multiple-word vocabulary items.

In hardware, this project has provided a useful testbed for recently-developed handheld and wearable computers.

Most significantly, this project has shown that commercially-available speech recognition technology has evolved to the point where it is operationally useful.

Implications for Further Research:

Since the start of this project, processors have become faster, memory has become less expensive, and large-vocabulary dictation systems for general English (for example, Dragon NaturallySpeaking) have become available as commercial products. Although no continuous recognizer yet has anything close to the elaborate menu tracking and scripting language found in DragonDictate, nonetheless the time has come to move to continuous speech-recognition technology. The hardware should be either more a powerful wearable or handheld platform, capable of doing large-vocabulary recognition in real time. Such machines, with 32 MB of RAM and a fast Pentium processor, are just becoming available. Doing so eliminates two major problems -- the inflexibility of the recognition vocabulary and the inappropriateness of the isolated-word fast match.

The time will come when systems like the Multilingual Interview System will include continuous speech recognition, fully automatic translation, and synthesized output speech in the target language. For systems that can be deployed in the next year, however, it is the opinion of the authors that research should focus on developing a system that uses state-of-the-art continuous speech recognition to control the playback of pre-recorded speech as in the present system.

INTRODUCTION

Traditionally, speech recognition systems have been classified as either "discrete" or "continuous." Most discrete systems have a vocabulary consisting of isolated words, used for dictation, and/or short phrases, used for command and control. A pause of at least 250 milliseconds is required between successive utterances. Continuous recognition systems are capable of recognizing sequences of words spoken without pause, perhaps under restrictions imposed by a grammar, or perhaps with probabilistic constraints imposed by a language model. Since 1990, Dragon Systems has had a product, named DragonDictate, that is capable of doing discrete recognition. Until April of 1997, with the announcement of Dragon NaturallySpeaking, there was no commercially-available dictation system for general English (although a number of excellent research systems that support continuous dictation have been developed under the sponsorship of DARPA and other Government agencies.

For recognition of sentences such as "Answer my question yes or no" or "I am a member of the NATO peacekeeping forces" (or "...the United States Army," "...Navy," or "...Marines,") either technology is appropriate. In the case where the total number of legal sentences is unlimited or even very large, only continuous recognition will suffice. However, for a system like the Multilingual Interview System, in which the total number of legal sentences is necessarily limited because their translations all need to be recorded, isolated-word technology is entirely appropriate. Even for the largest application module created by NOMI, the total number of legal items was only slightly more than 10,000, far less than DragonDictate's limit of 64,000 vocabulary items.

To implement speech recognition, then, all that was necessary was the replacement of the standard vocabulary of DragonDictate (60,000 isolated words plus commands) by a custom vocabulary consisting of recorded phrases like "Answer my question yes or no" or concatenations of recorded phrases like "I am a member of" ... "the NATO peacekeeping forces." Since the acoustic models for items to be recognized are based on sequences of phoneme codes, it was also comparatively straightforward to build new acoustic models for long sentences that rely on the same speaker-independent data files of phonetic models that are used in the DragonDictate product. Although building acoustic models "from scratch" for 10,000 sentences would have been a major undertaking, building such models from what was already part of the DragonDictate product was not especially time consuming. The techniques that were used, and the complications introduced by the need for so-called "rapid-match models," will be described in detail later in this report.

The usual "action" that a dictation system executes upon recognizing a word is to display that word on the computer's screen. However, the actions taken upon recogntion of a command are more diverse and complex, generally extending to almost anything that is supported by an application program or by the operating system, such as "move down five lines" or "simulate a double-click of the mouse." By using the scripting language that supports the definition of actions for DragonDictate commands, it was not hard to define appropriate actions like "play the wave file for 'I am a member' followed by the wave file for 'of the United States Navy.'" The techniques that were developed for automatically generation of such scripting files will also be described in detail later.

Because the utterances to be recognized are in general rather long, the recogntion task in the Multilingual Interview System is significantly easier than the general-English dictation task for which the DragonDictate product was developed. As a consequence, the system can function well with less adaptation to the user's voice than is normally required. However, limitations of the hardware on which the system was deployed introduced some unanticipated challenges.

In addition to recognizing spoken phrases, the Multilingual Interview System must interact with its user in a variety of ways, most of them related to the need to provide quick access to phrases that the user has not yet memorized and to provide a method for eliciting and recording responses. For the most part this is just straightforward Windows application programming. The main part of the report will discuss issues that are significant for the design of a module, such as categories, dialogs, and response codes.

The goal of the rest of this report is to describe the construction of the Multilingual Interview system in sufficient detail that the prospective developer of a new module could gain a clear idea of what tasks are involved and what tools are available to help with them. We will discuss the factors that make the speech-recognition task more or less difficult with the intent of providing guidance to a prospective developer, and we will identify issues related to system hardware that emerged as important during testing at Dragon System, at NOMI, at Fort Bragg, and in Bosnia.

METHODS, ASSUMPTIONS, AND PROCEDURES

Defining a Multilingual Interview System (MIS) Module:

An MIS module is specified in a text file whose format has been kept compatible with the original Lee Morin Medical Translator. Most lines in the file consist simply of a decimal number followed by some English text. The number, combined with a two-letter code for the chosen output language, determines what recording will be played. Numbers that end in ".00" denote "phrases" that are displayed in the main list of the application program, while numbers that end in ".01," ".02, ... denote subphrases that are displayed in a subsidiary list.

Preceding each group of lines is a category specification, identified as such by the fact that it does not begin with a digit. The application software builds a list of categories that can be displayed to the user. Either by speaking the name of a desired category or by clicking on it with a mouse, the user can cause the phrases in that category to be displayed.

By convention, the first line in the file is a text string that identifies the module. The application software reads the first line of each file in order to display to the user a list of available modules, from which the desired module can be selected by voice.

Any phrase or subphrase can be followed by a "jump code," which permits the user to jump to a set of related phrases elsewhere in the vocabulary and then jump back to the original position. Any question can be followed by a "response code" which, if the user has placed the system in "response mode," initiates an appropriate response dialog.

The following excerpt from the file "force.mis", which defines the force protection module, illustrates many of these features. The codes that are enclosed in ordinary parentheses are jump codes, while those enclosed in braces, like {n}, are response codes. The first line, "LANDMINE CHARACTERISTICS," is a category name.

```
LANDMINE CHARACTERISTICS

1108.00 Questions
1109.00 Do you know where any dangerous material of any kind is stored? {y}
1110.00 Did you see someone laying the landmines?{y}
1111.00 When did they do that? {n}
1112.00 Were they (military? civilian?) {c}
1112.01 military?
1112.02 paramilitary?
1112.03 civilian?
1113.00 Who were they?
1114.00 Where are they now?
1115.00 Have you seen a landmine explode there?{y}
1116.00 When did it explode? {n}
1117.00 Do you know anyone who has been hurt by a landmine? {y}
1118.00 Did you see it happen?{y}
1119.00 How many were hurt? {n}
1120.00 Do you know where the boundaries of the minefield are? {y}
1121.00 Can you point to the boundaries?{y}
1122.00 How far does the minefield extend?
1123.00 Is there an area you can walk through safely?{y}
1124.00 Have you walked there?{y}
1125.00 Have you seen others walk there?{y}
1126.00 How are the landmine locations marked?
```

1127.00 Can you draw what the markings look like?{y}
1128.00 Can you draw what the landmines look like?{y}

Customizing the Recognition Vocabulary:

In order to build a speech interface for the Multilingual Interview System, it is first necessary to customize the DragonDictate recognition vocabulary so that it includes all appropriate MIS vocabulary items. The process for modifying the recognition vocabulary in this manner is presented below.

Customizing the recognition vocabulary involved developing and using PERL scripts, in conjunction with Dragon in-house utilities and the DragonDictate Vocabulary Manager, to generate, format and incorporate appropriate acoustic and recognition data into DragonDictate program files.

Acoustic Data:

In order to modify the recognition vocabulary, appropriate acoustic data must be provided for each MIS vocabulary item. This data consists of a) phonetic pronunciations and b) word start assignments. Once these data are generated, they are added to the DragonDictate pronunciation vocabulary file and become available for use in building appropriate acoustic models.

Phonetic Pronunciations:

Dragon has developed a procedure to build appropriate pronunciations for all items and specified concatenations in an MIS module file ,based on an existing pronunciation database for over 200,000 individual words. Most words in the MIS vocabulary already exist in this pronunciation database. Those that do not, however, must be assigned a pronunciation manually and added to the database.

The Multilingual Interview System vocabulary contains many phrases. Pronunciations are built for these phrases by combining phonetic data for the individual words that make up the phrase. Since individual words in the phrases may have more than one pronunciation, however, it is important to use only appropriate word pronunciations when building the phrase pronunciations; failure to do so leads to a huge number of unnecessary alternate pronunciations, many of which do not reflect how a speaker would actually pronounce the phrase. The pronunciation-building procedure is designed to create pronunciations for a phrase that reflect how a speaker might actually say that phrase. The procedure therefore uses only appropriately stressed pronunciations when building phonetic models. In addition, pronunciations that are irrelevant or undesirable for a given domain are excluded based on a list compiled by the developer.

The pronunciation-building procedure requires the following data input:

1) A module file specifying the appropriate MIS vocabulary items for which pronunciations must be built
2) Dragon's existing database of pronunciations for more than 200,000 individual words, modified to include pronunciations for new words
3) Lists of pronunciations to add to and exclude from the above database while building pronunciations for a specific domain.

Word Start Assignments:

Once appropriate phonetic pronunciation models have been built for all MIS vocabulary items, each item is given an assignment a word-start group for use in building rapid-match models. Word start assignment is a completely automatic process, and requires as input only a list of words with their phonetic pronunciations.

Modifying DragonDictate:

Once each MIS item has a phonetic pronunciation and a word start assignment, this data is added to DragonDictate's pronunciation building dictionary file using an in-house tool for modifying DragonDictate program files. DragonDictate will use this data to build acoustic models for the MIS vocabulary items when they are later added to the DragonDictate recognition vocabulary file.

Recognition Data:

In order for DragonDictate to recognize MIS items, they must be added to the DragonDictate recognition vocabulary file with appropriate corresponding recognition data. This data is compiled in a DragonDictate DDX file, and imported into the DragonDictate recognition vocabulary file using the DragonDictate Vocabulary Manager.

The DDX file must contain the following information:

1) The DragonDictate vocabulary into which the data is being imported. This is the recognition vocabulary that is active whenever the MIS application itself is active.
2) The DragonDictate recognition group into which the data is being imported. Each MIS module has its own recognition group, which is available for recognition only when the corresponding MIS module is active.
3) The vocabulary items being added, and the DragonDictate or MIS application actions that are to be executed upon recognition of each item. The series of executed actions is the same for each MIS vocabulary item: the microphone is turned off, a keystroke sequence is sent to notify the application to play the appropriate .wav file or files, and the microphone is turned back on.

A DDX file specifying the correct recognition data for each MIS vocabulary item and concatenation is generated automatically with a PERL script. It is then incorporated into the DragonDictate recognition via the "import vocabulary" command in the Vocabulary Manager.

Below is an excerpt from an automatically generated DDX file:

```
switch-to-vocabulary MIS
switch-to-group /create FORCE
add-word "[I AM A MEMBER OF THE NATO PEACEKEEPING FORCES]" /script "
  SetMicrophone 0
  SendKeys  \"{alt+i}1080.00{Alt+3}{alt+i}1080.01{Alt+3}{alt+4}\"
  SetMicrophone 1
  " /nsc /nf
add-word "[I AM A MEMBER OF THE UNITED STATES AIR FORCE]" /script "
  SetMicrophone 0
  SendKeys  \"{alt+i}1080.00{Alt+3}{alt+i}1080.05{Alt+3}{alt+4}\"
  SetMicrophone 1
  " /nsc /nf
add-word "[I AM A MEMBER OF THE UNITED STATES ARMY]" /script "
  SetMicrophone 0
  SendKeys  \"{alt+i}1080.00{alt+3}{alt+i}1080.02{alt+3}{alt+4}\"
  SetMicrophone 1" /nsc /nf
add-word "[I AM A MEMBER OF THE]" /script "
  SetMicrophone 0
  SendKeys \"{alt+i}1080.00{Alt+3}{alt+4}\"
  SetMicrophone 1
" /nsc /nf
add-word "[LET US USE THIS MACHINE TO TALK TOGETHER]" /script "
  SetMicrophone 0
   SendKeys \"{alt+i}1.06{Alt+3}{alt+4}\"
  SetMicrophone 1
  " /nsc /nf
add-word "[I WANT TO ASK YOU QUESTIONS ABOUT MASS GRAVES]" /script "
  SetMicrophone 0
  SendKeys \"{alt+i}1141.00{Alt+3}{alt+4}\"
  SetMicrophone 1
```

```
" /nsc /nf
add-word "[I WILL BE BACK SOON]" /script "
  SetMicrophone 0
   SendKeys \"{alt+i}15.00{Alt+3}{alt+4}\"
  SetMicrophone 1
" /nsc /nf
add-word "[IS THIS THE PERSON YOU SAW]" /script "
  SetMicrophone 0
  SendKeys \"{alt+i}679.12{Alt+3}{alt+4}\"
  SetMicrophone 1
" /nsc /nf
add-word "[IS THIS WHAT YOU WISH TO SAY]" /script "
  SetMicrophone 0
   SendKeys \"{alt+i}2.00{Alt+3}{alt+4}\"
  SetMicrophone 1
" /nsc /nf
add-word "[IS THIS YOUR BAG]" /script "
  SetMicrophone 0
  SendKeys \"{alt+i}665.05{Alt+3}{alt+4}\"
  SetMicrophone 1
" /nsc /nf
```

Producing Wave Files for Voice Output:

Wave files that were produced for the initial deployment to Bosnia were recorded using a standard general-purpose recording program. Initial and final silence surrounding the recorded speech had to be removed manually – a tedious task when thousands of files need to be processed!

Since assuming responsibility for the Windows application software used in this project, Dragon Systems has developed a recording program specifically for the task of recording translated phrases, with the following features:

1) It reads a text file in the ".mis" format described above and prompts the user (either in English or in the target language) with the translated phrase to be spoken.
2) It automatically detects the start and end of speech and trims each file so that no more than about 200 milliseconds of silence either precedes or follows the speech.
3) It automatically adjusts the recording level to make use of a wide range of 16-bit digital sample values without any risk of clipping.
4) It normalizes each recording to use the full dynamic range that is permitted by the 8-bit linear format in which the recorded speech is saved.
5) It can automatically skip over files that already exist (during recording) or ones that do not exist (during playback).
6) It generates the conventional names for wave files from the language name (which is converted to a two-letter code) and the numeric code, and it places the file in the appropriate directory.

Voice-Enabling the User Interface:

DragonDictate's automatic menu and window tracking feature provided immediate speech support for multiple aspects of the MIS user interface, including support for many menu items, buttons, check boxes, radio buttons and static text captions as well as automatic tracking of vocabulary recognition groups. Those aspects of the user interface, however, that were either unsupported by DragonDictate, or supported in a way that was not optimal for military operation, were speech enabled with the DragonDictate scripting language. In some cases, appropriate scripting commands are communicated to DragonDictate directly from MIS application via Dynamic Data Exchange; in others, DDX files were used to add appropriate speech macros to the DragonDictate program files. Some factors that influence the implementation of the speech interface are presented below:

Mode of Operation:

NOMI has required that the Multilingual Interview System be operational in an eyes-free context, though much more emphasis was placed on this feature in the initial version of the software than in subsequent versions.

When a user is operating the Multilingual Interview System with a visual display, he or she can see on the screen what has been recognized and what actions have subsequently been executed. For eyes-free operation, however, it was necessary to find an alternative way of notifying the user of what has been recognized. For recognition of application commands such as "Switch to <Language>" and "Switch to <Categories>", the speech macros for the commands play a .wav file that notifies the user that the command has been recognized and executed. For recognition of MIS vocabulary phrases, a "Verification Mode" was developed that allows the user to first hear the recognized phrase played in English, then choose whether or not to play the translation for the recognized phrase. Though this Verification Mode is implemented in the MIS application code, it is accessed only via the speech interface, since eyes-free operation is possible only by voice control.

Overcoming DragonDictate Tracking Limitations:

DragonDictate does not automatically track edit control, list boxes or combo boxes. Since several important features, including Categories, Module, Language and Dialog Selection and Search, use such Windows controls, it was necessary to modify the speech interface to support these features. In cases where the list box always contains the same items, speech support was added for each item in the list box so that items could be selected by voice. For example, a user can select a category from the Categories list by saying the name of the category. In addition, commands were added for navigating in a list box by saying "Move <Direction> <Number>". If the items in a list box vary during use of the application, it is possible only to provide navigational support for the list box. For example, after using the Search function to compile a list of phrases, the user can select a phrase from the list only by using the "Move <Direction> <Number>" command.

It is important to note that for future customization of DragonDictate, Dragon's toolkit XTools will permit dynamic vocabulary creation at application runtime; thus, it will be possible to voice enable all list boxes, including those with variable contents.

DragonDictate automatically tracks active windows based on window captions and activates appropriate corresponding recognition vocabularies. In areas where automatic window tracking does not correctly set recognition vocabularies, however, vocabulary management was done via the DragonDictate scripting language. In some cases, scripting commands are sent to DragonDictate from the MIS application by using Dynamic Data Exchange. This method is used to set the recognition vocabulary for Modules, Categories, Dialogs and the Response Mode dialogs. In other cases, DragonDictate macros were used. This was done, for example, to enable alpha-bravo spelling for keystroke entry in various Windows File Save and Open dialogs.

Adapting Acoustic Models to the User:

The acoustic models used by DragonDictate include "rapid-match models" and "hidden Markov models." Both sets of models were developed by processing nearly 100 hours of transcribed recorded speech from a wide variety of talkers. In the DragonDictate product they are adapted whenever the user corrects an error, and many users configure the product to adapt the models on the basis of every word that is spoken.

The rapid-match models have the task of reducing the initial, very large, set of words in the vocabulary to a manageable number, typically about 1000 words for the case of a Pentium processor. This is done by making very simple models for several thousand word-start groups of words which sound alike in their first syllable or two. For example, all words that begin with "counter-" will belong to the same word-start group. In building the rapid-match models for DragonDictate, great pains were taken so construct the word-start groups in such a way that common words like "the" are modeled accurately, even if this means occasionally constructing a word-start group with only one member.

As the first step in recognition, the start of an utterance (about 400 milliseconds of speech) is compared with the models for all the word-start groups, and the members of those groups are then given more careful consideration. During adaptation, the speech data that was compared with the word-start models is used to improve the model that corresponds to the correct word. There is no possibility of moving a word from one word-start group to another or of creating a new word-start group if the model for the group to which the word was assigned turns out to have been inappropriate.

As the second step in recognition in the case where all words are assumed to be equally likely, the hidden Markov models for the words belonging to the best-scoring word-start groups are compared with the complete spoken utterance, and the best-matching model is selected. The hidden Markov models have been designed to capture variability among speakers, and for polysyllabic words and phrases they tend to give the correct answer even without any adaptation. Adaptation improves their performance, but for tasks like the MIS task, the improvement is scarcely necessary.

When DragonDictate is used for isolated-word dictation, almost every word that is spoken is one that was represented in the training data and was used in designing the rapid-match models. As a result, the rapid match models perform quite well at their task of reducing a set of 30,000 or 60,000 candidate words to 1000 words or so. As the models for the word-start groups are adapted, the word-start models become better , and this improves performance on the entire vocabulary. By the time a user has adapted on the basis of a couple of thousand word, the rapid-match models contribute very little to the error rate.

When DragonDictate is used for the MIS recognition task, many of the vocabulary items are phrases that begin with sequences of phonemes that are not found in any English word. These phrases are assigned to the best-matching word-start group, but the match is often not very good – for example, the word-start group for "do you know..." contains the word "Dukakis." As a result, the models occasionally fail at the apparently simple task of reducing a vocabulary of 6000 words to 1500 candidate hidden Markov models. Adaptation helps, but if the word-start assignment was a poor match to start with, it takes several spoken examples to change the model enough so that adaptation is effective. At the same time, the hidden Markov models perform their task very well, since they have to distinguish only among complete long phrases, not among short segments at the start of a word.

As a consequence, the DragonDictate acoustic models perform much better in the Multilingual interview system when the rapid-match step is unnecessary. This is the case when the vocabulary size is less than the number of hidden Markov models can process in real time – about 1500 items for an 80486 processor and about 6000 items for a fast Pentium processor. Under other circumstances, the rapid-match step is the major cause of recognition errors.

Results from recognition performance testing on two scripts, recorded by the same user, illustrate the impact of the rapid-match step on recognition accuracy. On script A, a 294-word script consisting of the entire new module developed by NOMI for PSYOPS operations, recognition accuracy was 86.73% when run without adaptation on a vocabulary of 6500 words, using a rapid-match threshold of 1500 words. This 6500 word vocabulary consisted of the entire Force Protection Module that was deployed in Bosnia, plus the new PSYOPS module. Recognition accuracy increased to %91.16 percent with adaptation. When the rapid-match threshold was set to 6000, thus almost completely eliminating the need for rapid-match models, accuracy increased to 99.62% on an unadapted user. Errors consisted of such misrecognitions as "collar" (recognized as "color") and "fairly bad" (recognized as "very bad"). With adaption, accuracy increased to 97.96%. On script B, which was based on phrases from MIS dialogs developed by NOMI, recognition accuracy was 89.63% when run without adaptation on the vocabulary of 6500 words, using a rapid-match threshold of 1500. With adaptation, accuracy increased to 92.98%. When the rapid-match threshold was increased to 6000, accuracy increased to 97.99% on an unadapted user file and to 99% on an adapted user.

Features that Support Two-way Communication:

The original Lee Morin Medical Translator supported two-way communication only by means of spoken phrases like "Hold up the number of fingers," "Squeeze my hand once for yes," or "Point to where it hurts." Questions like "What is your rank?" were accompanied my a list of possible responses which could be played in succession until the interviewee indicated that he had heard the correct answer.

In the Multilingual Interview system this approach has been codified in the form of "response dialogs": one for yes-no questions, one for numerical responses, and one for lists of answers to questions like "What is your rank?" Each dialog contains pushbuttons that play phrases like "Answer my question yes or no" or that make it easy to play possible responses in the interviewee's language. The plan is to add more such dialogs if they work well in practice.

Among features that have been discussed but not yet implemented are dialogs to present a map on the computer screen in conjunction with questions like "Can you show me on the map?" and to present a set of bitmap images among which the interviewee can choose by pointing. A more ambitious idea, one that would require substantial research and development as well as field testing, is to attempt to do limited-vocabulary speech recognition in the interviewee's language to get the answer to questions like "What is your rank?"

RESULTS AND DISCUSSION

Specifications of Deployed System:

1996 Deployment:

Hardware:	IBM ThinkPad, 90 MHz Pentium laptop, Andrea ANC-100 microphone with mute switch
Software:	Multilingual Interview System, Lee Morin version Customized DragonDictate for Windows, 2.0 In-house Power Edition

1997 Deployment:

Hardware:	CMU/Telxon TIAP 40486 100 MHz wearable CDI 100 MHz Pentium wearable Andrea ANC-100 microphone with mute switch
Software:	Multilingual Interview System v1.5, Dragon version Customized DragonDictate 2.52, In-house Power Edition

Factors that Affect Recognition Performance:

Quality of Audio Input:

The quality of the audio input is a significant factor in the performance of the Multilingual Interview System. Both the audio front end of the computers on which the software has been deployed and the choice of microphone and push-to-talk switch have had implications for speech.

In the first Bosnia deployment, IBM ThinkPads were used. The audio front end on the ThinkPads posed a serious problem for speech recognition because the ThinkPad inserted a segment of digital silence at the beginning of each utterance. Dragon solved this problem by developing a method of replacing the digital silence by with a sample of real silence.

In the second deployment, wearable computers were used. While the audio front end on the CDI wearables were excellent for speech recognition, the Telxon systems initially had such a high level of background noise that speech recognition performance on these machines was unacceptable. Telxon later installed heat shielding in the Telxons that brought the background noise level to an acceptable, though not ideal, level.

Choice of Microphone:

Microphone quality is an important factor in recognition performance. NOMI tested many microphones to find one that was suitable both for use in the field and for use with speech recognition. This was particularly a challenge for the first deployment because very few microphones worked properly on the IBM ThinkPads. To overcome this problem, Dragon built an amplifier that increased the audio signal received from the microphone to a level suitable for the speech recognition. The Andrea microphone was the only microphone to work without this amplifier.

Another microphone-related factor that has affected speech recognition is the use of a push-to-talk switch. The potentially high level of environmental noise in the field necessitated the use of a push-to-talk switch when using the s Interview System by voice. However, most push-to-talk switched introduced ambient noise that were misrecognized by DragonDictate as short words. Many different push-to-talk switches were tested, including one built by Dragon. Best results were obtained with the Andrea ANC-100, which comes with a mute switch. While ambient noise is not a problem with the Andrea, however, the use of the mute switch can have an adverse effect on recognition if the user begins speaking before having fully switched the microphone on.

Processor Speed and Vocabulary Size:

Because of the problem with rapid-match models described in the section above, both the processor speed of the deployed units and the vocabulary size of the MIS module have affected speech recognition. When running the Multilingual Interview System on a Pentium processor, rapid-match prefiltering can be disabled and thus recognition accuracy is almost perfect. On a 40486-based machine, however, it is not possible to disable prefiltering unless the vocabulary is sufficiently small (below 1500) so that disabling prefiltering does not make recognition run too slow.

The vocabulary size of the MIS module used in the second Bosnia deployment was close to 6000 (this figure includes alternate pronunciations for the same MIS vocabulary item). It was therefore not possible to disable prefiltering on the 40486 machines.

Experience with Systems in Bosnia:

A Dragon representative joined colleagues from NOMI in Sarajevo, Bosnia-Herzegovina to work with users in the field. Unforeseen shipping difficulties delayed the arrival of new wearable units, however, so only one Telxon unit was available for training and testing. Because of the known speech recognition difficulties on the Telxon, Dragon efforts in Sarajevo were focused more on the development of the MIS software application itself than on the speech interface.
Nonetheless, Dragon did gain some interesting feedback about voice control of the application.

Users definitely stressed the importance of the speech interface. Because the display on the wearables is very small, and the pen software can be awkward to use, all users felt that a reliable and accurate voice interface is essential to system efficiency and usability. One user stated that a reliable voice interface increases usefulness of the system by %70.

In addition, some feedback on the MIS application itself was also relevant to the voice interface. Users stated a preference for a system that is simple and easy to use. For example, rather than have one module with a large vocabulary available for translation, they would prefer a number of smaller modules, each containing only vocabulary items that were relevant to a particular task. In this manner, they could easily get familiar with those modules that were appropriate for their work. For purposes of the speech interface, it would be easier for them to memorize appropriate phrases. In addition, better modularization of the MIS would increase performance on the Telxon 40486-based machine because modules could easily be kept to around 1000 phrases.

In addition, users emphasized the need for enhanced two-way communication capabilities. They would also like to have the ability to make an audio recording of interviewee responses.

CONCLUSIONS

Given a computer with a good audio front end and a fairly fast Pentium processor, speech recognition provides a very satisfactory front end for the Multilingual Interview System. Errors are infrequent except under strongly adverse noise conditions, and the point-and-click interface provides a fallback position in an environment like a helicopter.

While speech recognition is probably the only technology that can provide random access to thousands of phrases in a system of this type, it appears unlikely that even experienced users can ever learn the entire vocabulary. Thus application features such as the use of categories and dialogs have proved very important. The full-screen display that supports these and lets the user see what relevant phrases are available is important.

The performance on the Telxon 80486-based handheld computers was comparatively disappointing because of rapid-match errors in the speech recognition. Although a customized isolated-word-recognition algorithm could perhaps be devised that would achieve high accuracy on the MIS task with only an 80486 processor, no straightforward modification of the DragonDictate product worked to our full satisfaction.

Eventually systems like the MIS will be supplanted by fully-automatic translation systems with speech synthesis in all target languages, but until the translation technology is mature and natural-sounding synthesized speech is available in all languages of operational importance, the approach of having native speakers translate and record speech seems the best way to deal with simple interviewing and screening tasks.

RECOMMENDATIONS

Our strongest recommendation for the next-generation interview system is to move to a more powerful speech-recognition engine like the one in Dragon NaturallySpeaking. This is capable of recognizing general English, spoken continuously, from a 30,000-word active vocabulary in real time. The system requirements of a 133-MHz Pentium processor and 32MB of RAM are met or surpassed by wearable computers that are already available or will be available within a few months.

The NaturallySpeaking engine uses a new rapid-match algorithm whose models are derived from the hidden Markov models. When it is used for dictation, there are very few rapid-match errors. We would in fact recommend using it in dictation mode, rather than with a grammar that allows just the set of legal phrases and sentences to be recognized. A strong statistical language model, rather than a list of legal utterances or a finite-state grammar, can then enforce the expectation that the user is expected to say something in the standard vocabulary. If, in spite of this language model, the system recognizes something that is not in the standard vocabulary , it can then attempt to determine (perhaps by interacting with the user) what translated phrase should be spoken, or it can reject the input as failing to match anything in the set of available recordings.

Our other recommendation is that new applications should be designed from the start to exploit the possibility of playing back sequences of recordings. Sentences should be broken up in a way that does not violate grammatical constraints of the target languages, and it should be possible to put rules into the application program to generate whatever order of playback (e.g. main verb comes last) is appropriate in the target language.

APPENDIX A: Excerpts from Data files

Excerpt from DDX file:

```
switch-to-vocabulary /create /module MIS MIS

switch-to-group /create FORCE
add-word "[ARE ANY MEMBERS OF YOUR IMMEDIATE FAMILY HERE WITH YOU]" /script "
  SetMicrophone 0
  SendKeys \"{alt+i}2000.22{Alt+3}{alt+4}\"
  SetMicrophone 1
" /nsc /nf
add-word "[ARE THE LANDMINES IN ANY PATTERN]" /script "
  SetMicrophone 0
  SendKeys \"{alt+i}1131.00{Alt+3}{alt+4}\"
  SetMicrophone 1
" /nsc /nf
add-word "[ARE THERE ANY ORPHANS WHO NEED OUR HELP]" /script "
  SetMicrophone 0
  SendKeys \"{alt+i}51.11{Alt+3}{alt+4}\"
  SetMicrophone 1
" /nsc /nf
add-word "[ARE THERE ANY TRIP-WIRES NEAR IT]" /script "
  SetMicrophone 0
  SendKeys \"{alt+i}1135.00{Alt+3}{alt+4}\"
  SetMicrophone 1
" /nsc /nf
add-word "[ARE THERE PEOPLE WHO WANT TO HELP US]" /script "
  SetMicrophone 0
  SendKeys  \"{alt+i}1166.00{Alt+3}{alt+i}1166.02{Alt+3}{alt+4}\"
  SetMicrophone 1
  " /nsc /nf
add-word "[ARE THERE PEOPLE WHO WANT TO HURT US]" /script "
  SetMicrophone 0
  SendKeys  \"{alt+i}1166.00{Alt+3}{alt+i}1166.03{Alt+3}{alt+4}\"
  SetMicrophone 1
  " /nsc /nf
add-word "[ARE THERE PEOPLE WHO WANT TO STOP US]" /script "
  SetMicrophone 0
  SendKeys  \"{alt+i}1166.00{Alt+3}{alt+i}1166.01{Alt+3}{alt+4}\"
  SetMicrophone 1
  " /nsc /nf
add-word "[ARE THERE PEOPLE WHO WANT TO WORK WITH US]" /script "
  SetMicrophone 0
  SendKeys  \"{alt+i}1166.00{Alt+3}{alt+i}1166.04{Alt+3}{alt+4}\"
  SetMicrophone 1
  " /nsc /nf
add-word "[ARE THERE PEOPLE WHO WANT TO]" /script "
  SetMicrophone 0
  SendKeys \"{alt+i}1166.00{Alt+3}{alt+4}\"
  SetMicrophone 1
" /nsc /nf
add-word "[ARE THERE WIRES OR HORNS PROTRUDING FROM IT]" /script "
```

```
  SetMicrophone 0
  SendKeys \"{alt+i}1134.00{Alt+3}{alt+4}\"
  SetMicrophone 1
" /nsc /nf
add-word "[ARE THEY ARMED WITH EXPLOSIVES]" /script "
  SetMicrophone 0
  SendKeys \"{alt+i}667.04{Alt+3}{alt+4}\"
  SetMicrophone 1
" /nsc /nf
add-word "[ARE THEY ARMED WITH FIREARMS]" /script "
  SetMicrophone 0
  SendKeys \"{alt+i}667.03{Alt+3}{alt+4}\"
  SetMicrophone 1
" /nsc /nf
add-word "[ARE THEY ARMED]" /script "
  SetMicrophone 0
  SendKeys \"{alt+i}667.02{Alt+3}{alt+4}\"
  SetMicrophone 1
" /nsc /nf
add-word "[ARE THEY MARKED]" /script "
  SetMicrophone 0
  SendKeys \"{alt+i}1140.00{Alt+3}{alt+4}\"
  SetMicrophone 1
" /nsc /nf
add-word "[ARE YOU A LOCAL CIVILIAN]" /script "
  SetMicrophone 0
  SendKeys \"{alt+i}2140.00{Alt+3}{alt+4}\"
  SetMicrophone 1
" /nsc /nf
add-word "[ARE YOU A MEMBER OF THE ARMED FORCES]" /script "
  SetMicrophone 0
  SendKeys \"{alt+i}2000.19{Alt+3}{alt+4}\"
  SetMicrophone 1
" /nsc /nf
" /nsc /nf
add-word "[CAN WE BUY A MEAL]" /script "
  SetMicrophone 0
  SendKeys \"{alt+i}686.00{alt+3}{alt+i}686.06{alt+3}{alt+4}\"
  SetMicrophone 1
" /nsc /nf
add-word "[CAN WE BUY ASPHALT FOR ROADS]" /script "
  SetMicrophone 0
  SendKeys \"{alt+i}686.00{alt+3}{alt+i}686.03{alt+3}{alt+4}\"
  SetMicrophone 1
" /nsc /nf

add-word "[WRITE DOWN YOUR FULL UNIT DESIGNATION]" /script "
  SetMicrophone 0
  SendKeys \"{alt+i}2136.00{Alt+3}{alt+4}\"
  SetMicrophone 1
" /nsc /nf
add-word "[WRITE DOWN YOUR PHONE NUMBER]" /script "
  SetMicrophone 0
  SendKeys \"{alt+i}680.04{Alt+3}{alt+4}\"
  SetMicrophone 1
" /nsc /nf
add-word "[WRITE IT DOWN]" /script "
  SetMicrophone 0
  SendKeys \"{alt+i}2032.00{Alt+3}{alt+4}\"
  SetMicrophone 1
" /nsc /nf
add-word "[WRITE THE NUMBER ON THIS PAPER PLEASE]" /script "
  SetMicrophone 0
   SendKeys \"{alt+i}24.00{Alt+3}{alt+4}\"
  SetMicrophone 1
" /nsc /nf
add-word "[WRITE THEM DOWN]" /script "
  SetMicrophone 0
  SendKeys \"{alt+i}2033.00{Alt+3}{alt+4}\"
  SetMicrophone 1
" /nsc /nf
add-word [WRITE] /script "
  SetMicrophone 0
  SendKeys \"{alt+i}780.10{Alt+3}{alt+4}\"
  SetMicrophone 1
" /nsc /nf
add-word [WRITING] /script "
  SetMicrophone 0
  SendKeys \"{alt+i}2060.04{Alt+3}{alt+4}\"
  SetMicrophone 1
" /nsc /nf
add-word [YEARS] /script "
  SetMicrophone 0
   SendKeys \"{alt+i}17.04{Alt+3}{alt+4}\"
  SetMicrophone 1
" /nsc /nf
add-word [YELLOW] /script "
  SetMicrophone 0
   SendKeys \"{alt+i}1330.08{Alt+3}{alt+4}\"
  SetMicrophone 1
" /nsc /nf
add-word "[YES OR NO]" /script "
  SetMicrophone 0
   SendKeys \"{alt+i}11.00{Alt+3}{alt+4}\"
  SetMicrophone 1
" /nsc /nf
add-word [yes?] /script "
  SetMicrophone 0
 SendKeys \"{alt+i}11.01{alt+3}{alt+4}\"
 SetMicrophone 1"
add-word [YESTERDAY] /script "
  SetMicrophone 0
   SendKeys \"{alt+i}18.02{Alt+3}{alt+4}\"
  SetMicrophone 1
" /nsc /nf
add-word [YET] /script "
  SetMicrophone 0
  SendKeys \"{alt+i}38.15{Alt+3}{alt+4}\"
  SetMicrophone 1
" /nsc /nf
add-word "[YOU ARE A PRISONER]" /script "
  SetMicrophone 0
  SendKeys \"{alt+i}1296.09{Alt+3}{alt+4}\"
  SetMicrophone 1
" /nsc /nf
add-word "[YOU ARE TRESPASSING HERE]" /script "
  SetMicrophone 0
  SendKeys \"{alt+i}681.05{Alt+3}{alt+4}\"
  SetMicrophone 1
" /nsc /nf
add-word "[YOU CANNOT ENTER THIS AREA]" /script "
  SetMicrophone 0
  SendKeys \"{alt+i}669.01{Alt+3}{alt+4}\"
  SetMicrophone 1
" /nsc /nf
add-word "[YOU CANNOT GO THROUGH UNTIL I SEARCH YOUR PROPERTY]" /script "
  SetMicrophone 0
  SendKeys \"{alt+i}665.04{Alt+3}{alt+4}\"
  SetMicrophone 1
" /nsc /nf
add-word "[YOU DON'T KNOW]" /script "
  SetMicrophone 0
   SendKeys \"{alt+i}34.07{Alt+3}{alt+4}\"
  SetMicrophone 1
" /nsc /nf
add-word "[YOU MAY BE SHOT IF YOU ENTER THIS AREA ILLEGALLY]" /script "
  SetMicrophone 0
  SendKeys \"{alt+i}670.03{Alt+3}{alt+4}\"
  SetMicrophone 1
" /nsc /nf
add-word "[YOU MIGHT ONLY SEE A SMALL PART ABOVE GROUND]" /script "
  SetMicrophone 0
   SendKeys \"{alt+i}1593.00{Alt+3}{alt+4}\"
  SetMicrophone 1
" /nsc /nf
add-word "[YOU MUST BE ABLE TO GIVE THE LESSON BY YOURSELF]" /script "
  SetMicrophone 0
   SendKeys \"{alt+i}731.01{Alt+3}{alt+4}\"
  SetMicrophone 1
" /nsc /nf
add-word "[YOU MUST BE ABLE TO IDENTIFY DEMOLITION EQUIPMENT]" /script "
  SetMicrophone 0
   SendKeys  \"{alt+i}742.01{Alt+3}{alt+i}742.02{Alt+3}{alt+4}\"
  SetMicrophone 1
  " /nsc /nf
add-word "[YOU MUST BE ABLE TO MAINTAIN DEMOLITION EQUIPMENT]" /script "
  SetMicrophone 0
   SendKeys  \"{alt+i}742.01{Alt+3}{alt+i}742.03{Alt+3}{alt+4}\"
  SetMicrophone 1
  " /nsc /nf
add-word "[YOU MUST LEAVE YOUR BAG HERE]" /script "
  SetMicrophone 0
  SendKeys \"{alt+i}665.01{Alt+3}{alt+4}\"
  SetMicrophone 1
" /nsc /nf

switch-to-group /create "Dialog Control"
add-word "[<Move> <Direction Up Down> <Number/1 to 40>]" /script "
  SendKeys \"{\"+Move_1+Direction_Up_Down_1+\" \"+_1_to_40_1+\"}\",1"
add-word [directory] /script "
  SendKeys \"{Alt+i}\"" /nsc /nf
add-word [done] /script "
  SetMicrophone 0
  PlaySound \"c:\\mis\\cues\\done.wav\"
  SetMicrophone 1
  SendKeys \"{alt+d}\"
" /nsc /nf
add-word [edit] /keys "{alt+e}{down 2}" /nsc
add-word [expand] /keys {alt+x} /nsc
add-word [jump] /script "
  SendKeys \"{alt+j}\""
add-word [last] /script "
  SendKeys \"{alt+a}\"
" /nsc /nf
add-word [less] /script "
  SendKeys \"{alt+l}\"" /nsc /nf
add-word [locate] /script "
  SendKeys \"{alt+c}\"" /nsc
add-word [more] /keys {alt+m} /nsc /nf
add-word [next] /script "
  SendKeys \"{Alt+t}\"
" /nsc /nf
add-word "[page down]" /keys {ExtPgDn} /nsc
add-word "[page up]" /keys {ExtPgUp} /nsc
add-word [play] /script "
  SendKeys \"{Alt+p}\"
" /nsc /nf
add-word [prev] /script "
  SendKeys \"{alt+r}\"
/nsc /nf
add-word [previous] /script "
  SendKeys \"{alt+r}\"
add-word "[scroll down]" /keys {ExtDown} /nsc
add-word "[scroll up]" /keys {ExtUp} /nsc
add-word [select] /keys {alt+g} /nsc
add-word [skip] /script "
  SendKeys \"{Alt+s}\"
" /nsc /nf
add-word "[verify next]" /script "
  SendKeys \"{down}{alt+v}\"
" /nsc /nf
add-word [verify] /script "
  SendKeys \"{alt+f}\"
" /nsc /nf

switch-to-group /create "Play the Translation?"
```

```
add-word [Cancel] /script "
  SetMicrophone 0
  PlaySound \"c:\\mis\\cues\\canceled.wav\"
  SetMicrophone 1
  ControlPick \"Cancel\"
  SendKeys \"{Alt+6}\"
" /nsc /nf
add-word [OK] /script "
  SetMicrophone 0
  ControlPick \"OK\"
  SendKeys \"{Alt+5}\"
  SetMicrophone 1" /nsc /nf

switch-to-group /create "Dialog Directory"
add-word "[<Move> <Direction Up Down> <Number/1 to 40>]" /script "
  SendKeys \"{\"+Move_1+Direction_Up_Down_1+\" \"+_1_to_40_1+\"}\",1"
add-word "[Actions to Take When a Mine is Found]" /script "
   SendKeys \"{alt+1}\"
   Wait 300
   SendKeys \"Actions to Take When a Mine is Found{alt+2}\"
" /nsc /nf
add-word [Cancel] /script "
  ControlPick \"Cancel\"
  Wait 300
  ResetGroup" /nsc /nf
add-word "[Civilian Landmine Intro]" /script "
   SendKeys \"{alt+1}\"
   Wait 300
   SendKeys \"Civilian Landmine Intro{alt+2}\"
" /nsc /nf
add-word "[Clarification and Building Parts]" /script "
   SendKeys \"{alt+1}\"
   Wait 300
   SendKeys \"Clarification and Building Parts{alt+2}\"
" /nsc /nf
add-word "[Coalition Resource Interview]" /script "
  SendKeys \"{Alt+1}\"
  Wait 300
  SendKeys \"Coalition Resource Interview{Alt+2}\"
" /nsc /nf
add-word "[Construct a Nonelectric Initiating Detonating Assembly]" /script "
   SendKeys \"{alt+1}\"
   Wait 300
   SendKeys \"Construct a Nonelectric Initiating{alt+2}\"" /nsc /nf
add-word "[Destroy Booby Traps]" /script "
   SendKeys \"{alt+1}\"
   Wait 300
   SendKeys \"Destroy Booby Traps{alt+2}\"
" /nsc /nf
add-word "[Destroy Mines and UXO]" /script "
   SendKeys \"{alt+1}\"
   Wait 300
   SendKeys \"Destroy Mines and UXO{alt+2}\"
" /nsc /nf
```

```
add-word "[Detection of Landmines]" /script "
   SendKeys \"{alt+1}\"
   Wait 300
   SendKeys \"Detection of Landmines{alt+2}\"
" /nsc /nf
add-word "[Direct a De-mining Team]" /script "
   SendKeys \"{alt+1}\"
   Wait 300
   SendKeys \"Direct a Demining Team{alt+2}\"" /nsc /nf
add-word [done] /script "
  SetMicrophone 0
  PlaySound \"c:\\mis\\cues\\done.wav\"
  SetMicrophone 1
  SendKeys \"{alt+d}\"
" /nsc /nf
add-word [expand] /script "
  SendKeys \"{alt+x}\"" /nsc
add-word "[Explain Translator Ground Rules to Patient]" /script "
   SendKeys \"{alt+1}\"
   Wait 300
   SendKeys \"Explain Translator Ground Rules to Patient{alt+2}\"
" /nsc /nf
add-word "[Geographic Features]" /script "
   SendKeys \"{alt+1}\"
   Wait 300
   SendKeys \"Geographic Features{alt+2}\"
" /nsc /nf
add-word "[Harsh Checkpoint Challenge]" /script "
  SendKeys \"{Alt+1}\"
  Wait 300
  SendKeys \"Harsh Checkpoint Challenge{Alt+2}\"" /nsc /nf
add-word "[Humanitarian Aid Offer]" /script "
  SendKeys \"{Alt+1}\"
  Wait 300
  SendKeys \"Humanitarian Aid Offer{Alt+2}\"
" /nsc /nf
add-word "[Instructor Teach Back]" /script "
   SendKeys \"{alt+1}\"
   Wait 300
   SendKeys \"Instructor Teach Back{alt+2}\"
" /nsc /nf
add-word "[Introduction and Identification]" /script "
  SendKeys \"{Alt+1}Introduction and Identification{Alt+2}\"
" /nsc /nf
add-word "[page down]" /keys {ExtPgDn} /nsc
add-word "[page up]" /keys {ExtPgUp} /nsc
```

Excerpt for .MIS file:

FORCE PROTECTION AND AREA SECURITY

INTRODUCTION

1.00 Do you speak...<this language>? (Customary greeting/etc.){y}
1.01 Customary greeting
1.02 What's your country? {e}
1.03 What's the capital? {e}
1.04 Do you know who I am? {y}
1.05 Do you know where you are? {y}
1.06 Let us use this machine to talk together
2.00 Is this what you wish to say? {y}
3.00 Answer my question yes or no (Do what I ask)
3.01 Do what I ask
4.00 Put up the required number with your fingers {n}
5.00 Good morning, did you sleep well? {y}
6.00 Good afternoon
7.00 Good evening
8.00 Good night, sleep well
9.00 Good bye
10.00 How are you? (Fine?/Better?/Not too good?/Not too bad?) {10}
10.01 Fine?
10.02 Better?
10.03 Not too good?
10.04 Not too bad?
11.00 Yes--No (Yes?/No?/Squeeze my hand for yes/etc.) {11}
11.01 Yes?
11.02 No?
11.03 Yes!
11.04 No!
11.05 Raise your hand if you understand
11.06 Squeeze my hand once for yes
11.07 Squeeze my hand twice for no
11.08 Move your toes
11.09 Please repeat
12.00 Please (Thank you)
12.01 Thank you
13.00 What do you want?
14.00 I will get it for you
15.00 I will be back soon (In a few/Just a moment)
15.01 In a few
15.02 Just a moment
16.00 Seconds (Minutes/Hours)
16.01 Minutes
16.02 Hours
17.00 Days (Nights/Weeks/etc.)
b17.01 Nights
17.02 Weeks
17.03 Months
17.04 Years
18.00 Today (Tomorrow/Yesterday)
18.01 Tomorrow
18.02 Yesterday
19.00 In the morning (At noon/In the evening/At night)
19.01 At noon
19.02 In the evening
19.03 At night

REGISTRATION AND IDENTIFICATION

50.00 What is your nationality? (2047 nationalities) {2047}
51.00 What is your name? (My name is.../What is your family name?/etc.) {51}
51.01 My name is...
51.02 What is your family name?
51.03 What is your street name?
51.04 What is your nickname?
51.05 What name do people call you?
51.06 Where do you stay now?
51.07 When will you return home?
51.08 Are you married?{y}
51.09 Do you have children?{y}
51.10 Is your family safe today?{y}
51.11 Are there any orphans who need our help?{y}
51.12 What is the greatest need among the local people here?
52.00 Please show me your... (Identity card/Identity disc/Dog tags){c}
52.01 Identity card
52.02 Identity disc
52.03 Dog tags
53.00 What languages do you speak? (2047 languages) {2047}
54.00 Please print in capital letters here (Name/Service number/DOB/NOK/etc.) {c}
54.01 Your surname (last name)
54.02 Your first name
54.03 Your service/regimental number {n}
54.04 Your date of birth
54.05 Your place of birth
54.06 Your home address
54.07 How old are you? {n}
54.08 Information about next of kin
54.09 What relation are you? {c}
54.10 To him?
54.11 To her